Adaptive Knowledge Propagation in Web Ontologies

Pasquale Minervini, Claudia d'Amato, Nicola Fanizzi, Floriana Esposito

[1] Department of Computer Science - University of Bari, Italy
{firstname.lastname}@uniba.it

Abstract. The increasing availability of structured machine-processable knowledge in the WEB OF DATA calls for machine learning methods to support standard reasoning based services (such as query-answering and logic inference). Statistical regularities can be efficiently exploited to overcome the limitations of the inherently incomplete knowledge bases distributed across the Web. This paper focuses on the problem of predicting missing class-memberships and property values of individual resources in Web ontologies. We propose a transductive inference method for inferring missing properties about individuals: given a class-membership/property value learning problem, we address the task of identifying relations which are likely to link similar individuals, and efficiently propagating knowledge across such (possibly diverse) relations. Our experimental evaluation demonstrates the effectiveness of the proposed method.

1 Introduction

Standard query answering and reasoning services for the Semantic Web [2] (SW) largely rely on deductive inference. However, purely deductive reasoning with SW representations suffers from several limitations: inference tasks can be computationally complex, and distributed knowledge bases (KBs) are often characterized by incomplete and conflicting knowledge. In this context, many complex tasks (such as query answering, clustering or ranking) are ultimately based on assessing the truth value of ground facts. Deciding on the truth of specific facts (assertions) in SW knowledge bases requires to take into account the *open-world* form of reasoning adopted in this context: a failure on ascertaining the truth of a given fact does not imply that such fact is false, but rather that its truth value cannot be deductively inferred from the KB (e.g. because of a temporary lack of knowledge). This differs from the *Negation As Failure*, commonly used with databases and logic programs. Other issues are related to the distributed nature of the data across the Web. Cases of contradictory answers or flawed inferences may be caused by distributed pieces of knowledge that may be mutually conflicting.

The prediction of the truth value of an assertion can be cast as a *classification* problem to be solved through *statistical learning*: individual resources in an ontology can be regarded as statistical units, and their properties can be statistically inferred even when they cannot be deduced from the KB. Several approaches have been proposed in the SW literature (see [15] for a recent survey). A major issue with the methods proposed so far is that the induced statistical models (as those produced by kernel methods, tensor factorization, etc.) are either difficult to interpret by experts and to integrate in logic-based SW infrastructures, or computationally impractical.

Related Work A variety of methods have been proposed for predicting the truth value of assertions in Web ontologies: those include generative models [16], kernel methods [12], upgrading of propositional algorithms [11], matrix and tensor factorization methods [14, 19]. An issue with existing methods is that they either rely on a possibly expensive search process, or induce statistical models that are often not easy to interpret by human experts. Kernel methods induce models (such as separating hyperplanes) in a high-dimensional feature space implicitly defined by a kernel function. The underlying kernel function itself usually relies on purely syntactic features of the neighborhood graphs of two individual resources (such as their common subtrees [12] or isomorphic subgraphs [21]). In both cases, there is not necessarily a direct translation in terms of domain knowledge. Latent variable and matrix or tensor factorization methods such as [14, 16, 19] try to explain the observations in terms of latent classes or attributes, which also may be non-trivial to describe using the domain's vocabulary. The approach in [11] tries to overcome this limitation by making use of complex features defined using the ontology's terminology; however, this method involves a search process in a possibly very large feature space, which might not be feasible in practice.

Contribution We propose a transductive inference method for predicting the truth value of assertions, which is based on the following intuition: individuals that are *similar* in some aspects tend to be linked by specific relations. Yet it may be not straightforward to determine such relations for a given learning task. Our approach aims at identifying such relations, and permits the efficient propagation of information through chains of related individuals. It turns out to be especially useful with real-world *shallow* ontologies, i.e. those with a relatively simple, fixed terminology and populated by very large amounts of instance data such as citation or social networks, in which related entities tend to influence each other. These are particularly frequent in the context of the *Linked Open Data* [7] (LOD).

Unlike other approaches, the proposed method can be used to identify which relations are likely to link examples with similar characteristics. Similarly to graph-based semi-supervised learning (SSL) methods [5], we rely on a similarity graph linking pairs of similar examples, for propagating knowledge among them. SSL methods are often designed for propositional representations, while the proposed method addresses the problem of learning from real ontologies, where examples (represented by individuals in the KB) can be interlinked by diverse relations. In particular, this article makes the following contributions:

- A method for efficiently *propagating* knowledge among similar examples: it leverages a similarity graph, which plays a critical role in the propagation process.
- An approach to *learning* an optimal similarity graph for a given prediction task, by leveraging a set of semantically diverse relations among examples in the ontology.

To the best of our knowledge, our approach is the first to explicitly identify relations that semantically encode similarities among examples w.r.t. a given learning task. The method proposed in this article is a significant advance w.r.t. our previous work in [13], in which we adopt kernel-defined weights to construct the similarity graph. However, such weights were lacking a meaningful interpretation.

The remainder of the paper is organized as follows. In Sect. 2, we review the basics of semantic knowledge representation and reasoning tasks, and we introduce the concept of *transductive learning* in the context of semantic KBs. In Sect. 3, we illustrate the proposed method, which is based on the efficient propagation of information among related examples, and address the problem of identifying which relations are likely to link similar examples. In Sect. 4, we provide empirical evidence for the effectiveness of the proposed method. In Sect. 5, we summarize the proposed approach, outline its limitations and discuss possible future research directions.

2 Transductive Learning with Web Ontologies

We assume the knowledge base (KB) is encoded in a syntactic variant of some *Description Logic* [1] (DL), and describes a set of objects, their attributes and relations. Basics elements are *atomic concepts* $N_C = \{C, D, \ldots\}$ interpreted as subsets of a domain of objects (e.g. `Person` or `Article`), and *atomic roles* $N_R = \{R, S, \ldots\}$ interpreted as binary relations on such a domain (e.g. `friendOf` or `authorOf`). Domain objects are represented by *individuals* $N_I = \{a, b, \ldots\}$, each associated to a domain entity (such as a person in a social network, or an article in a citation network).

Specifically, we consider KBs in the OWL 2 language [1], which has its theoretical foundations in DLs: concepts and roles are referred to as *classes* and *properties*, respectively. Classes, properties and individuals are represented in the ontology by their URIs. Each DL provides a set of constructors that can be used to build complex concept descriptions using atomic concepts and roles.

A DL KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ is composed by two main components: a *TBox* $\mathcal{T}$, which contains terminological axioms, and an *ABox* $\mathcal{A}$, which contains ground axioms (called *assertions*) about individuals. In the following, we will denote as $\mathsf{Ind}(\mathcal{A})$ the set of individuals occurring in $\mathcal{A}$.

As inference procedure, *Instance Checking* consists in deciding whether $\mathcal{K} \models Q(a)$ (where Q is a query concept and a is an individual) holds. Because of the *Open-World Assumption*, instance checking may provide three possible outcomes, i.e. i) $\mathcal{K} \models Q(a)$, ii) $\mathcal{K} \models \neg Q(a)$ and iii) $\mathcal{K} \not\models Q(a) \land \mathcal{K} \not\models \neg Q(a)$. This means that failing to deductively infer the membership of an individual a to a concept Q does not imply that a is a member of its complement $\neg Q$.

It is also possible to express more complex queries: given a (infinite) set of variables N_V, a *Conjunctive Query* q is a conjunction of concept or role atoms ($C(v)$ or $R(v, v')$, with $v, v' \in N_V \cup N_I$) built on the signature of $\mathcal{K}$. The set of its variables $\textsc{Var}(q)$ is composed by *answer variables* and (existentially) *quantified variables*. Informally, a binding of the variables w.r.t. some model of $\mathcal{K}$ determines the satisfiability of a query and a result via the answer variables values. $\mathcal{K} \models q$ denotes the satisfiability of q w.r.t. all models of $\mathcal{K}$.

In this work, we focus on *transductive learning* [20] rather than *inductive learning*. Inductive learning focuses on the creation of general rules, which are then applied to test cases, while transductive learning focuses on generalizing directly from observed training cases to specific test cases.

[1] OWL 2 W3C Recommendation: `http://www.w3.org/TR/owl-overview/`

The main motivation behind the choice of transductive learning is described by the *main principle* in [20]: "If you possess a restricted amount of information for solving some problem, try to solve the problem directly and never solve a more general problem as an intermediate step. It is possible that the available information is sufficient for a direct solution but is insufficient for solving a more general intermediate problem".

On the ground of the available information, the method proposed in this work aims at learning a *labeling function* for a given target class that can be used for predicting whether examples, represented by individuals in the knowledge base, are members of a target class C (positive class) or to its complement $\neg C$ (negative class), when this cannot be deductively inferred. This setting is closely related to the *transductive classification* setting in e.g. [22].

Formally, the problem can be defined as follows:

Definition 2.1 (Transductive Class-Membership Learning).
Given:

- *a* target *class C in a KB $\mathcal{K}$;*
- *a set of examples $X \subseteq \mathsf{Ind}(\mathcal{A})$ partitioned into:*
  - *a set of* positive examples*:* $X_+ \triangleq \{a \in X \mid \mathcal{K} \models C(a)\}$;
  - *a set of* negative examples*:* $X_- \triangleq \{a \in X \mid \mathcal{K} \models \neg C(a)\}$;
  - *a set of* neutral *(unlabeled)* examples*:* $X_0 \triangleq \{a \in X \mid a \notin X_+ \wedge a \notin X_-\}$;
- *a set of* labeling functions $\mathcal{F}$ *with domain X and range* $\{-1, +1\}$*, i.e.*

$$\mathcal{F} \triangleq \{\mathbf{f} \mid \mathbf{f} : X \to \{+1, -1\}\};$$

- *a* cost function $\mathrm{cost}(\cdot) : \mathcal{F} \mapsto \mathbb{R}$ *defined over labeling functions in $\mathcal{F}$;*

Find $\mathbf{f}^* \in \mathcal{F}$ *minimizing* $\mathrm{cost}(\cdot)$ *w.r.t. X:*

$$\mathbf{f}^* \leftarrow \arg\min_{\mathbf{f} \in \mathcal{F}} \mathrm{cost}(\mathbf{f}).$$

The transductive learning task is cast as the problem of finding a *labeling function* $\mathbf{f}^*$ for a target class C, defined over a finite set of *labeled* (if positive or negative) and *unlabeled* (if their membership to the target class cannot be determined) examples X, which minimizes an arbitrary cost criterion. The set of examples X is a subset of the individuals occurring in the KB.

Example 2.1 (Transductive Class-Membership Learning). Consider an ontology modeling an academic domain. The problem of learning whether a set of researchers is affiliated to a given research group or not, provided a set of positive and negative examples of affiliates, can be cast as a *transductive class-membership learning* problem: examples (consisting in a subset of the individuals in the ontology, each corresponding to a researcher), represented by the set X, can be either *positive*, *negative* or *neutral* depending on their membership to a target class `ResearchGroupAffiliate`. The transductive learning problem reduces to finding the best labeling function $\mathbf{f}$ (according to a given criterion, represented by the cost function), providing a membership value for each example in X.

In this work, we leverage the diverse relations holding among examples in the ontology to *propagate* knowledge, in the form of label information, among similar examples.

The method proposed in this article is related to *graph-based semi-supervised learning* [5] (SSL) methods. In particular, it is based on the *cluster assumption*: if two examples are in the same cluster, then their class memberships should be similar. Similarly to graph-based SSL methods, we define a similarity graph over examples, and look for a labeling function $\mathbf{f}$ that *varies smoothly* across similar examples (i.e. those linked together in the similarity graph).

3 Knowledge Propagation

In this section we present a new method, named *Adaptive Knowledge Propagation* (AKP), for solving the learning problem in Def. 2.1 in the context of Web ontologies. In Sect. 3.1 we show how a (weighted) similarity graph defined over examples can be efficiently used to propagate label information among similar examples. The effectiveness of this approach strongly depends on the choice of the similarity graph (represented in the following by its adjacency matrix $\mathbf{W}$). In Sect. 3.2, we show how the matrix $\mathbf{W}$ can be learned from examples, by leveraging their relationships within the ontology.

3.1 Transductive Inference as an Optimization Problem

We now propose a solution to the transductive learning problem in Def. 2.1. As discussed in the end of Sect. 2, we look for a labeling function $\mathbf{f}^*$ defined over examples X, which is both consistent with labeled examples, and *varies smoothly* across examples in the same cluster (according to the cluster assumption). In the following, we assume that a (weighted) similarity graph over examples in X is already provided. Such a graph is represented by its adjacency matrix $\mathbf{W}$, such that $\mathbf{W}_{ij} = \mathbf{W}_{ji} \geq 0$ if $x_i, x_j \in X$ are *similar*, and 0 otherwise. As in [5, ch. 11], we assume that $\mathbf{W}_{ii} = 0$. A solution to the problem of learning $\mathbf{W}$ from examples is proposed in Sect. 3.2.

Formally, each labeling function $\mathbf{f}$ can be represented by a finite-size vector, where $\mathbf{f}_i \in \{-1, +1\}$ is the label for the i-th element in the set of examples X. According to [22], labels can be enforced to vary smoothly among similar examples by considering a cost function with the following form:

$$E(\mathbf{f}) \triangleq \frac{1}{2} \sum_{i=1}^{|X|} \sum_{j=1}^{|X|} \mathbf{W}_{ij} (\mathbf{f}_i - \mathbf{f}_j)^2 + \epsilon \sum_{i=1}^{|X|} \mathbf{f}_i^2, \qquad (1)$$

where the first term enforces the labeling function to vary smoothly among similar examples (i.e. those connected by an edge in the similarity graph), and the second term is a L_2 regularizer (a penalty on the *complexity* of the labeling function) with weight $\epsilon > 0$ over $\mathbf{f}$. A labeling for unlabeled examples in X_0 is obtained by minimizing the function $E(\cdot)$ in Eq. 1, constraining the value of $\mathbf{f}_i$ to 1 (resp. -1) for all positive examples $x_i \in X_+$ (resp. negative examples $x_i \in X_-$).

Let $L \triangleq X_+ \cup X_-$ and $U \triangleq X_0$ represent labeled and unlabeled examples, and $\mathbf{f}_L$, $\mathbf{f}_U$ their labels respectively. Constraining the labeling function $\mathbf{f}_U$ to take only discrete values on unlabeled examples (i.e. $\mathbf{f}_i \in \{-1, +1\}, \forall x_i \in X_0$) has two main drawbacks:

– The labeling function $\mathbf{f}$ can only provide a *hard* classification (i.e. $\mathbf{f}_U \in \{-1, +1\}^{|U|}$), without any measure of confidence;
– The function $E(\cdot)$ defines the energy function of a discrete Markov Random Field, and calculating the marginal distribution over labels $\mathbf{f}_U$ is inherently difficult [10].

To overcome these problems, in [22] authors propose a continuous relaxation of $\mathbf{f}_U$, where labels for unlabeled examples are represented by real values (i.e. $\mathbf{f}_U \in \mathbb{R}^{|U|}$). This allows for a simple, closed-form solution to the problem of minimizing the function $E(\cdot)$ for a given value of $\mathbf{f}_L$, where $\mathbf{f}_L$ represents the labels for labeled examples.

Application to Class-Membership Learning. We can solve the learning problem in Def. 2.1 by minimizing the cost function $E(\cdot)$ in Eq. 1, for a given labeling for labeled examples $\mathbf{f}_L$. Eq. 1 can be rewritten as [22]:

$$E(\mathbf{f}) = \mathbf{f}^T(\mathbf{D} - \mathbf{W})\mathbf{f} + \epsilon\mathbf{f}^T = \mathbf{f}^T(\mathbf{L} + \epsilon\mathbf{I})\mathbf{f}, \tag{2}$$

where $\mathbf{D}$ is a diagonal matrix such that $\mathbf{D}_{ii} = \sum_{j=1}^{|X|} \mathbf{W}_{ij}$ and $\mathbf{L} \triangleq \mathbf{D} - \mathbf{W}$ is the *graph Laplacian* of $\mathbf{W}$. Reordering the vector $\mathbf{f}$ and matrices $\mathbf{W}$ and $\mathbf{L}$ w.r.t. the membership of examples to L and U, they can be rewritten as:

$$\mathbf{f} = \begin{bmatrix}\mathbf{f}_L \\ \mathbf{f}_U\end{bmatrix}, \quad \mathbf{W} = \begin{bmatrix}\mathbf{W}_{LL} & \mathbf{W}_{LU} \\ \mathbf{W}_{UL} & \mathbf{W}_{UU}\end{bmatrix}, \quad \mathbf{L} = \begin{bmatrix}\mathbf{L}_{LL} & \mathbf{L}_{LU} \\ \mathbf{L}_{UL} & \mathbf{L}_{UU}\end{bmatrix}. \tag{3}$$

The problem of finding a real valued labeling function $\mathbf{f}_U$ which minimizes the cost function $E(\cdot)$ for a given value of $\mathbf{f}_L$ has a closed form solution [22]:

$$\mathbf{f}_U^* = (\mathbf{L}_{UU} + \epsilon\mathbf{I})^{-1}\mathbf{W}_{UL}\mathbf{f}_L. \tag{4}$$

Complexity. A solution for Eq. 4 can be computed efficiently in nearly-linear time w.r.t. $|X|$. Indeed computing $\mathbf{f}_U^*$ can be reduced to solving a linear system in the form $\mathbf{A}\mathbf{x} = \mathbf{b}$, with $\mathbf{A} = (\mathbf{L}_{UU} + \epsilon\mathbf{I})$, $\mathbf{b} = \mathbf{W}_{UL}\mathbf{f}_L$ and $\mathbf{x} = \mathbf{f}_U^*$. A linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$ with $\mathbf{A} \in \mathbb{R}^{n \times n}$ can be solved in nearly linear time w.r.t. n if the coefficient matrix $\mathbf{A}$ is *symmetric diagonally dominant*[2] (SDD). An algorithm for solving SDD linear system is proposed in [9]: its time-complexity is $\approx \mathrm{O}\big(m \log^2 n\big)$, where m is the number of non-zero entries in $\mathbf{A}$ and n is the number of variables in the system of linear equations. In Eq. 4, the matrix $(\mathbf{L}_{UU} + \epsilon\mathbf{I})$ is SDD, since the graph Laplacian $\mathbf{L}$ is SDD [18].

3.2 Learning to Propagate Knowledge in Web Ontologies

The approach to propagate knowledge across similar examples discussed in Sect. 3.1 relies on a similarity graph, represented by its adjacency matrix $\mathbf{W}$.

The underlying assumption of this work is that some relations among individuals in the KB might encode a similarity relation w.r.t. a specific target property or class: identifying such relations can help to propagate information among similar examples, and provide new knowledge about the domain.

[2] A matrix $\mathbf{A}$ is SDD iff $\mathbf{A}$ is symmetric (i.e. $\mathbf{A} = \mathbf{A}^T$) and $\forall i : \mathbf{A}_{ii} \geq \sum_{i \neq j} |\mathbf{A}_{ij}|$.

In literature, this phenomenon is also referred to as *Homophily* [3]: a relation between examples (such as *friendship* in a social network) can be correlated with those individuals being similar w.r.t. a set of properties (such as political views, hobbies, religious beliefs). However, depending on the learning task at hand, not all relations are equally effective at encoding similarities between examples. For example, in a social network, friends may tend to share common interests, while quiet people may tend to prefer talkative friends and vice-versa.

In this work, we represent each distinct relation type by means of an *adjacency matrix* $\tilde{\mathbf{W}}$, such that $\tilde{\mathbf{W}}_{ij} = \tilde{\mathbf{W}}_{ji} = 1$ iff the relation $\texttt{rel}(x_i, x_j)$ between x_i and x_j holds in the ontology (i.e. $\mathcal{K} \models \texttt{rel}(x_i, x_j)$). The predicate `rel` might represent any generic relation between examples (e.g. friendship or co-authorship). For simplicity, we assume $\tilde{\mathbf{W}}_{ii} = 0, \forall i$.

Given a set of adjacency matrices $\mathcal{W} \triangleq \{\tilde{\mathbf{W}}_1, \ldots, \tilde{\mathbf{W}}_r\}$, according to the assumption that not all relations are equally important in the construction of the similarity graph, we define $\mathbf{W}$ as a linear combination of the matrices in $\mathcal{W}$:

$$\mathbf{W} \triangleq \sum_{i=1}^{r} \mu_i \tilde{\mathbf{W}}_i, \quad \text{with } \mu_i \geq 0, \forall i \tag{5}$$

where each μ_i, is a parameter representing the contribution of the matrix $\tilde{\mathbf{W}}_i$ in the construction of $\mathbf{W}$. Non-negativity in $\boldsymbol{\mu}$ enforces that $\mathbf{W}$ has non-negative weights, and therefore that the corresponding graph Laplacian $\mathbf{L}$ is PSD [18]. This ensures that the problem of finding a global minimum for Eq. 2 has a unique solution that can be calculated in closed form. In the following, we propose a solution to the problem of efficiently learning the parameters $\{\boldsymbol{\mu}, \epsilon\}$ from a set of labeled and unlabeled examples.

Parameters Learning. The parametric form of $\mathbf{W}$ is fully specified by the parameters $\boldsymbol{\mu}$ in Eq. 5, which reflect the importance of each relation in the construction of the similarity graph. In addition, the approach in Sect. 3.1 depends on the choice of a regularization parameter ϵ. In this work, we propose learning the parameters $\boldsymbol{\Theta} \triangleq \{\boldsymbol{\mu}, \epsilon\}$ by *Leave-One-Out* (LOO) *Error minimization*. Provided that propagation can be performed efficiently, we are able to directly computing the LOO error: it is defined as the summation of reconstruction errors obtained by considering each labeled example, in turn, as unlabeled, and predicting its label.

Let $U_i \triangleq U \cup \{x_i\}$ and $L_i \triangleq L - \{x_i\}$: w.l.o.g. we assume that the label of the left-out example $x_i \in L$ is in the first position of the new real valued labeling vector $\mathbf{f}_{U_i}$. Let $\ell(x, \hat{x})$ be a generic, differentiable loss function (e.g. $\ell(x, \hat{x}) = |x - \hat{x}|$ for the absolute loss, or $\ell(x, \hat{x}) = (x - \hat{x})^2/2$ for the quadratic loss). The LOO Error is formally defined as follows:

$$\mathcal{Q}(\boldsymbol{\Theta} \mid \mathbf{f}_L) \triangleq \sum_{i=1}^{|L|} \ell(\mathbf{f}_i, \hat{\mathbf{f}}_i), \tag{6}$$

where $\mathbf{e}^T \triangleq (1, 0, \ldots, 0) \in \mathbb{R}^{u+1}$ and $\hat{\mathbf{f}}_i \triangleq \mathbf{e}^T(\mathbf{L}_{U_iU_i} + \epsilon\mathbf{I})^{-1}\mathbf{W}_{U_iL_i}\mathbf{f}_{L_i}$ represents the continuous label value assigned to x_i as if such a value was not known in advance.

The vector $\mathbf{e}^T$ is needed to select the predicted label for the left-out example $x_i \in L$. This leads to the definition of the following criterion for learning the parameters $\boldsymbol{\Theta}$:

Definition 3.1 (Minimum LOO Error Parameters). *Given a set of labeled (resp. unlabeled) examples L (resp. U) and a set of adjacency matrices $\mathcal{W}$, each corresponding to a relation type, the* minimum LOO Error Parameters $\boldsymbol{\Theta}^*_{LOO}$ *are defined as follows:*

$$\boldsymbol{\Theta}^*_{LOO} = \arg \min_{\substack{\boldsymbol{\Theta}=\{\boldsymbol{\mu},\epsilon\} \\ \boldsymbol{\mu}\geq \mathbf{0},\epsilon>0}} \mathcal{Q}(\boldsymbol{\Theta} \mid \mathbf{f}_L) + \lambda ||\boldsymbol{\Theta}||^2, \quad (7)$$

where the function $\mathcal{Q}$ is defined as in Eq. 6 and $\lambda \geq 0$ weights a L_2 regularization term which controls the complexity of parameters $\boldsymbol{\Theta}$.

The objective function in Def. 3.1 is differentiable and can be efficiently minimized by using gradient-based function minimization approaches such as gradient descent. Let $\mathbf{Z}_i = (\mathbf{L}_{U_i U_i} + \epsilon \mathbf{I})$: the gradient of $\mathcal{Q}$ w.r.t. a parameter $\theta \in \boldsymbol{\Theta}$ is given by:

$$\frac{\partial \mathcal{Q}(\boldsymbol{\Theta} \mid \mathbf{f}_L)}{\partial \theta} = \sum_{i=1}^{|L|} \frac{\partial \ell(\mathbf{f}_i, \hat{\mathbf{f}}_i)}{\partial \hat{\mathbf{f}}_i} \left[\mathbf{e}^T \mathbf{Z}_i^{-1} \left(\frac{\partial \mathbf{W}_{U_i L_i}}{\partial \theta} \mathbf{f}_{L_i} - \frac{\partial \mathbf{Z}_i}{\partial \theta} \mathbf{f}^*_{U_i} \right) \right]. \quad (8)$$

3.3 Identifying Meaningful Relations

In Sect. 3.2, we expressed the adjacency matrix of the similarity graph $\mathbf{W}$ as a linear combination of adjacency matrices $\tilde{\mathbf{W}}_i \in \mathcal{W}$, each corresponding to a distinct relation type among examples within the ontology (such as friendship or co-authorship). We now discuss the problem of efficiently retrieve possibly non-trivial, yet meaningful, relations among examples in X.

Simply retrieving binary relations encoded by atomic roles in the KB might fail to capture a number of meaningful, semantically relevant, relations among examples. For example, in an academic domain, the KB may contain each researcher's group affiliations (e.g. by means of a `affiliatedTo` atomic role), but the relation "sharing the same affiliation" between researchers might not be captured by any atomic role between examples in X. To overcome this issue, we rely on Conjunctive Queries (CQs), described in Sect. 2, to retrieve relations between examples. For example, the relation "sharing the same affiliation" between $a, b \in X$ can be assessed by means of the following CQ:

$$? : \exists z.(\texttt{affiliatedTo}(a, z) \wedge \texttt{affiliatedTo}(b, z)),$$

where $z \in N_V$ is a *non-distinguished* variable.

In this work, we leverage the relations between examples holding in the KB for constructing the similarity graph; in particular, we consider relations that can be expressed using CQs. However, the number of such relations might be very large. To overcome this problem, in empirical evaluations (see Sect. 4), we considered two types of such relations holding between pairs of examples $a, b \in X$:

– *Simple* relations, i.e. those encoded by CQs in the form:

$$? : \texttt{r}(a, b),$$

where $\texttt{r} \in N_R$ is an atomic role;

– *Composite* relations, i.e. those corresponding to CQs in the form:

$$?:\exists z.(\mathtt{r}(a,z) \wedge \mathtt{r}(b,z)) \quad \text{or} \quad ?:\exists z.(\mathtt{r}(z,a) \wedge \mathtt{r}(z,b)),$$

where $\mathtt{r} \in N_R$ and $z \in N_V$.

Current technologies allow to efficiently retrieve complex relations holding among examples. CQs can be expressed in the SPARQL-DL [17] query language. SPARQL-DL seems particularly convenient for the task: it is a specialization of SPARQL, sharing its syntax and working under OWL's *Direct Model-Theoretic Semantics*[3]. SPARQL-DL queries generalize CQs as they admit variables standing for property names (allowing to retrieve different types of complex relations among individuals at once) together with *non-distinguished variables*, i.e. those that are bound to entities that need not be interpreted as specific individuals in the queried ontology.

Example 3.1. Given a KB $\mathcal{K}$, suppose that statistical units (i.e. individuals of interest) are all persons, represented by members of the concept $\mathtt{foaf:Person}$. Assume that relations of interest correspond to the result of CQs in the form:

$$?:\exists z.(\mathtt{foaf:Person}(a) \wedge \mathtt{foaf:Person}(b) \wedge \mathtt{r}(a,z) \wedge \mathtt{r}(b,z)),$$

where $\mathtt{r} \in N_R$, $a, b \in N_I$ and $z \in N_V$. Such relations can be retrieved by a single SPARQL-DL query:

```
SELECT DISTINCT ?p ?q ?r WHERE {
 ?p a foaf:Person .
 ?q a foaf:Person .
 ?p ?r _:o .
 ?q ?r _:o .
 ?r rdf:type owl:ObjectProperty .
}
```

Note that the variable `_:o` is a *non-distinguished variable* which does not need to be materialized in the KB (i.e. represented by an asserted individual). ■

This approach to retrieving complex relations presents multiple advantages: a single SPARQL-DL query can capture a large class of CQs, thanks to the use of variables in place of role names. We will use the following short-hand notations to describe more concisely relations elicited during the empirical evaluation phase:

$$\mathtt{rel}_1 \circ \mathtt{rel}_2^{-1}(a,b) \equiv \exists z.(\mathtt{rel}_1(a,z) \wedge \mathtt{rel}_2(z,b)),$$
$$\mathtt{rel}_1^{-1} \circ \mathtt{rel}_2(a,b) \equiv \exists z.(\mathtt{rel}_1(z,a) \wedge \mathtt{rel}_2(b,z)),$$

where $\mathtt{rel}_1, \mathtt{rel}_2 \in N_R$, $a, b \in X$ and $z \in N_V$.

3.4 Summary of the Proposed Method

The method, which we refer to as *Adaptive Knowledge Propagation* (AKP), can be summarized by the following steps:

[3] `http://www.w3.org/TR/owl2-direct-semantics`

1. Retrieve relations among examples in X using SPARQL-DL queries, and create a set of adjacency matrices $\mathcal{W} = \{\tilde{\mathbf{W}}_1, \ldots, \tilde{\mathbf{W}}_r\}$, one for each relation type.
2. Given a labeling for labeled examples $\mathbf{f}_L$, find the minimum Leave-One-Out Error parameters $\boldsymbol{\Theta}^*_{LOO}$ defined in Def. 3.1, by solving the constrained optimization problem in Eq. 7 (e.g. by using a gradient-based optimization method).
3. Use the learned parameters $\boldsymbol{\Theta}^*_{LOO} = \{\boldsymbol{\mu}, \epsilon\}$ to find a labeling for unlabeled examples $\mathbf{f}_U$, by first calculating the adjacency matrix of the similarity graph $\mathbf{W}$ as in Eq. 5, and then propagating knowledge across the graph as in Eq. 4.

4 Empirical Evaluation

The method discussed in Sect. 3 was experimentally evaluated in comparison with other approaches proposed in the literature on a variety of assertion prediction problems. Sources and datasets for reproducing the experiments are available at `https://code.google.com/p/akp/`. We now describe the setup of experiments and their outcomes.

4.1 Setup

In empirical evaluations, we used an open source DL reasoner [4] for answering the SPARQL-DL queries.

Ontologies We considered three real world ontologies: the AIFB PORTAL Ontology [5], the DBPEDIA 3.9 Ontology [4] and the BRITISH GEOLOGICAL SURVEY (BGS) Ontology [6]. The characteristics of these ontologies are outlined in Tab. 1.

The AIFB PORTAL Ontology relies on knowledge from the SWRC Ontology and metadata available from the Semantic Portal of the AIFB institute: it models key concepts within a research community, such as persons, articles, technical reports, projects and courses (e.g. $\sim$500 individuals belong to the class `foaf : Person` and $\sim$2400 to the class `foaf : Document`). DBPEDIA [4] makes structured information extracted from Wikipedia available in the LOD cloud, providing unique identifiers for the described entities that can be dereferenced over the Web: DBPEDIA 3.9, released in September 2013, describes 4.0 million entities. The BRITISH GEOLOGICAL SURVEY Ontology is part of an effort by the British Geological Survey, a premier center for earth science, to publish geological data (such as hydro-geological, gravitational and magnetic data). In particular, the ontology models knowledge on 11697 “Named Rock Units”.

Experimental Setting The proposed method, denoted AKP, is summarized in Sect. 3.4. We used *Projected Gradient Descent* to minimize the Leave-One-Out Error in Eq. 6 w.r.t. available labels $\mathbf{f}_L$ (using the absolute loss as loss function), together with an intermediate line search to assess the step size and early stopping. The regularization parameter λ in Eq. 6 was fixed to $\lambda = 10^{-8}$, preventing the parameters to diverge.

[4] Pellet v2.3.1 – `http://clarkparsia.com/pellet/`
[5] `http://www.aifb.kit.edu/web/Wissensmanagement/Portal`
[6] `http://data.bgs.ac.uk/`, as of March 2014

Table 1: Ontologies considered in the experiments

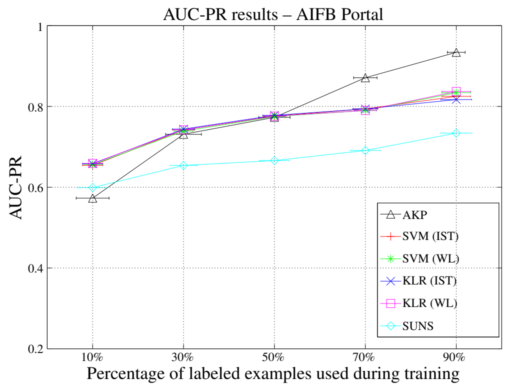
Ontology	DL Language	#Axioms	#Individuals	#Properties	#Classes
AIFB PORTAL [12]	$\mathcal{ALEHO}(\mathcal{D})$	268540	44328	285	49
DBPEDIA 3.9 [4] Frag.	$\mathcal{ALCH}$	78795	16606	132	251
BGS [21]	$\mathcal{ALI}(\mathcal{D})$	825133	87555	154	6

Before each experiment, the class-membership relations that were the target of the prediction task were removed from the ontology. Following the related evaluation procedures in [12, 21], members of the target class were considered as *positive examples*, while an equal number of *negative examples* was randomly sampled from unlabeled examples. Remaining instances (i.e. neither positive nor negative) were considered as *neutral* (unlabeled) *examples*. Results are reported in terms of *Area Under the Precision-Recall Curve* (AUC-PR), a measure to evaluate rankings also used in e.g. [14]. In each experiment, we considered the problem of predicting the membership to each of several classes: for each of such classes, we performed a 10-fold cross validation (CV), and report the average AUC-PR obtained using each of the considered methods.

We used the same 10-folds partitioning across experiments related to each of the datasets. For such a reason, we report statistical significance tests using a paired, non-parametric difference test (Wilcoxon T test). We also report diagrams showing how using a smaller random sample of labeled training examples (i.e. $10\%, 30\%, 50\%, \ldots$, a plausible scenario for a number of real world settings with limited labeled training data), and using the remaining examples for testing, affects the results in terms of AUC-PR.

Setup of the Compared Methods We compare AKP with state-of-the-art approaches proposed for learning from ontological KBs. Specifically, we considered two kernel methods: Soft-Margin SVM (SM-SVM) and Kernel Logistic Regression (KLR), together with different kernel functions suited for ontological KBs: we used the *Intersection SubTree* [12] (IST) and the *Weisfeiler-Lehman* [21] (WL) kernels for ontological KBs. We also considered the SUNS [19] relational prediction model.

The RDF graph used by kernel functions and SUNS was materialized as follows: all $\langle \mathtt{s}, \mathtt{p}, \mathtt{o} \rangle$ triples were retrieved by means of SPARQL-DL queries (where p was either an object or a data-type property) together with all *direct type* and *direct sub-class* relations. As in [12], IST kernel parameters were selected in $d \in \{1, 2, 3, 4\}$ and $\lambda_{ist} \in \{0.1, 0.3, \ldots, 0.9\}$, and WL kernel parameters in $d, h \in \{1, 2, 3, 4\}$ (where d represents the depth of the considered neighborhood graph). In SM-SVM, in order to obtain a ranking among instances, we applied the logistic function s to the decision boundary f instead of the sign function (which is commonly used in the classification context), thus obtaining $s(f(\cdot)) : \mathcal{X} \to [0, 1]$. In SM-SVM, the parameter C was selected in $C \in \{0.0, 10^{-6}, 10^{-4}, \ldots, 10^{4}, 10^{6}\}$, while in KLR the weight λ_k associated to the L_2 regularizer was selected in $\lambda_k \in \{10^{-4}, 10^{-3}, \ldots, 10^{4}\}$. In the SUNS relational prediction model, parameters t and λ were selected in $t \in \{2, 4, 6, \ldots, 24\}$ and $\lambda_s \in \{0, 10^{-2}, 10^{-1}, \ldots, 10^{6}\}$. All parameters were selected by grid optimization, using a 10-fold cross validation (CV) within the training set.



Method	AUC-PR (mean $\pm$ var.)	
AKP	0.932 ± 0.013	
SUNS	0.734 ± 0.030	▼
SMSVM (IST)	0.825 ± 0.025	▼
SMSVM (WL)	0.834 ± 0.025	▼
KLR (IST)	0.817 ± 0.029	▼
KLR (WL)	0.837 ± 0.025	▼

Fig. 1: AIFB PORTAL – Left: AUC-PR results (mean, std.dev.) estimated by 10-fold CV, obtained varying the percentage of labeled examples used for training – Right: AUC-PR results estimated by 10-fold CV: $\blacktriangledown/\triangledown$ (resp. $\blacktriangle/\triangle$) indicates that AKP's mean is significantly higher (resp. lower) in a paired Wilcoxon T test with $p < 0.05$ / $p < 0.10$

4.2 Results

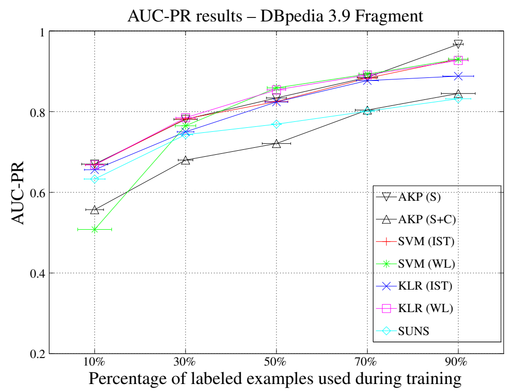
Experiments with the AIFB PORTAL Ontology. As in [12,21], the learning task consisted in predicting the affiliations of AIFB staff members to research groups. Specifically, in a set of 316 examples (each representing a researcher in the ontology), the task consisted in predicting missing affiliations to 5 distinct research groups. Empirical results are described in Fig. 1. The table (right) summarizes the overall AUC-PR results on the research group affiliation prediction task, obtained via 10-fold CV (one per research group, in a *one-versus-all* setting). The plot shows average AUC-PR values describes results obtained with a limited number of labeled training examples, and leaving the rest to the test: error bars (pictured horizontally) represent twice the standard deviation.

The proposed method AKP yields significantly higher AUC-PR values in comparison with the other considered methods, where statistical significance was calculated with a Wilcoxon T test with $p < 0.05$. By identifying those relations that are likely to link persons with similar research group affiliations, it shows that AKP can be used to discover new knowledge about the domain of an ontology. Tab. 2 shows a sample of the relations considered for the affiliation prediction task, among a total of 77 retrieved (all *composite*) relations, together with a measure of their relevance (given by their associated weight μ_i, described as either LOWER if $\mu_i \approx 0$, and HIGHER otherwise). As expected, AKP recognizes that authors sharing publications or interests, teaching the same courses or sharing their office are very likely to be affiliated to the same research group (unlike e.g. sharing the same academic title).

In this experiment, each AKP run took an average of ~ 500 seconds on a single core of an Intel®Core$^{\text{TM}}$i7 processor, showing that it can be used in practice for learning from real KBs.

Table 2: Relations considered in the AIFB PORTAL and the DBPEDIA 3.9 Ontologies

AIFB PORTAL		DBPEDIA 3.9	
HIGHER μ_i	LOWER μ_i	HIGHER μ_i	LOWER μ_i
`publications`$^{-1}$ $\circ$ `publications`	`title` $\circ$ `title`$^{-1}$	`vicePresident`	`successor`
`interest` $\circ$ `interest`$^{-1}$	`mobile` $\circ$ `mobile`$^{-1}$	`president`	`predecessor`
`lecturer`$^{-1}$ $\circ$ `lecturer`	`road` $\circ$ `road`$^{-1}$	`region` $\circ$ `region`$^{-1}$	`religion` $\circ$ `religion`$^{-1}$
`room` $\circ$ `room`$^{-1}$	`webpage` $\circ$ `webpage`$^{-1}$	`district` $\circ$ `district`$^{-1}$	`award` $\circ$ `award`$^{-1}$



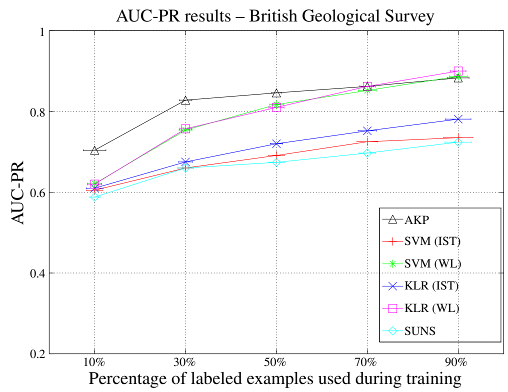
Method	AUC-PR (mean ± var.)	S	S+C
AKP (S)	0.967 ± 0.003		▲
AKP (S+C)	0.845 ± 0.035	▼	
SUNS	0.832 ± 0.019	▼	
SMSVM (IST)	0.930 ± 0.011	▽	
SMSVM (WL)	0.930 ± 0.011	▽	
KLR (IST)	0.888 ± 0.029	▼	
KLR (WL)	0.927 ± 0.012	▽	

Fig. 2: DBPEDIA 3.9 Ontology – Left: AUC-PR results (mean, st.d.) estimated by 10-fold CV, obtained varying the percentage of labeled examples used for training – Right: AUC-PR results estimated by 10-fold CV: ▼/▽ (resp. ▲/△) indicates that AKP's mean is significantly higher (resp. lower) in a paired Wilcoxon T test with $p < 0.05$ / $p < 0.10$

Experiments with the DBPEDIA 3.9 Fragment Similarly to [14], we evaluated the proposed approach on the task of predicting political party affiliations to either the Democratic and the Republican party for 82 US presidents and vice-presidents from DBPEDIA 3.9. The experiment illustrated in [14] uses a small RDF fragment containing the `president` and `vicePresident` predicates only.

In this experiment, we used a real-life fragment of DBPEDIA 3.9 obtained by means of a crawling process, containing a number of irrelevant and possibly noisy entities and relations. Following the extraction procedure in [8], the DBPEDIA 3.9 RDF graph was traversed starting from resources representing US presidents and vice-presidents: all immediate neighbors, i.e. those with a recursion depth of 1, were retrieved, together with their related schema information (direct classes and their super-classes, together with their hierarchy). All extracted knowledge was used to create a KB whose characteristics are outlined in Tab. 1. For efficiency reasons, parameters in the WL kernel were fixed to $d = 1$ and $h = 1$.

In this experiment, the total number of retrieved relations (both *simple* and *composite*) was higher than the number of instances itself: 82 US presidents and vice-presidents were interlinked by 25 simple relations and 149 composite relations. This differs from other empirical evaluations discussed in this paper, in which instances are linked by a



Method	AUC-PR (mean ± var.)	
AKP	0.887 ± 0.014	
SUNS	0.724 ± 0.022	▼
SMSVM (IST)	0.735 ± 0.026	▼
SMSVM (WL)	0.887 ± 0.010	
KLR (IST)	0.781 ± 0.020	▼
KLR (WL)	0.900 ± 0.007	

Fig. 3: BGS Ontology – Left: AUC-PR results (mean, st.d.) estimated by 10-fold CV, obtained varying the percentage of labeled examples used for training – Right: AUC-PR results estimated by 10-fold CV: $\blacktriangledown/\triangledown$ (resp. $\blacktriangle/\triangle$) indicates that AKP's mean is significantly higher (resp. lower) in a paired Wilcoxon T test with $p < 0.05$ / $p < 0.10$

more limited number of, exclusively composite, relations. For such a reason, we evaluated two variants of the proposed method: AKP (S), which only uses simple relations, and AKP (S+C), which uses both simple and composite relations.

Experimental results are summarized in Fig. 2. AUC-PR values obtained with AKP (S) are significantly higher than those provided by the other considered methods ($p < 0.05$, except for three cases in which $p < 0.10$). This was not true for AKP (S+C): relying on both simple and composite relations greatly increased the variance in AUC-PR results. An explanation for this phenomenon is in the *curse of dimensionality* [6]: as the number of considered relations grows, it becomes increasingly difficult to identify those that effectively encode similarities among examples.

AKP was able to identify which relations are likely to link same party affiliates, some of which are summarized in Tab. 2: it was able to find that the vice president is likely to belong to the same party of the president; that representatives covering a role under the same president are likely to belong to the same party; or that representatives elected in the same region are likely to belong to the same party. On the other hand, AKP recognized that sharing the same religion, profession, nationality or awards does not necessarily mean sharing the same party affiliation.

Experiments with the British Geological Survey Ontology. As in [21], we evaluated AKP on the *Lithogenesis* prediction problem in the British Geological Survey Ontology. The problem consisted in predicting the value of the property `hasLithogenesis` in a set of 159 named rock units labeled with their corresponding lithogenetic type. As in [21], we focus on two learning tasks, consisting in the prediction of two major lithogenetic types: "Alluvial" and "Glacial".

Results are summarized in Fig. 3. AKP provides significantly higher AUC-PR values, in comparison with kernel methods using the IST kernel and SUNS ($p < 0.05$). The difference between results obtained with AKP and by kernel methods using the WL kernel was not statistically significant, confirming the effectiveness of the WL kernel on this specific dataset (see [21]). However, while the statistical models produced with the WL kernel only have non-trivial geometrical interpretations, those learned by AKP explicitly represent the importance of relations in the propagation process.

Also in this case, AKP was able to extract relations between rock units that are likely to link rocks with similar lithogenetic types. For example, among a total of 23 relations (all *composite*) it emerged that rocks with similar geographical distributions, thickness and lithological components were likely to share their lithogenetic type, while their geological theme and oldest geological age were not considered informative.

5 Conclusions and Future Work

In this work, we propose a semi-supervised transductive inference method for statistical learning in the context of the WEB OF DATA. Starting from the assumption that some relations among examples in a Web ontology can encode similarity information w.r.t. a given prediction task (pertaining a particular property of examples, such as a specific class-membership), we propose a method, named *Adaptive Knowledge Propagation*, for i) identifying which relations are likely to link similar examples in the ontology, and ii) efficiently propagating knowledge across related examples, leveraging the diverse nature of such relations.

We empirically show that the proposed method is able to identify which relations are likely to link examples that are similar w.r.t. a given aspect, and that this information can provide new knowledge about the ontology. We also show that AKP provides significantly better or competitive results, in terms of AUC-PR, in comparison with current state-of-the-art methods in the literature. We are currently investigating probabilistic ways of learning how to propagate knowledge among examples; the use of different loss functions, optimization methods and regularization terms; and the automatic identification and selection of more complex relations between examples.

Acknowledgments. This work fulfills the objectives of the PON 02_00563_3489339 project "PUGLIA@SERVICE - Internet-based Service Engineering enabling Smart Territory structural development" funded by the Italian Ministry of University and Research (MIUR).

References

1. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F. (eds.): The Description Logic Handbook. Cambridge University Press (2007)
2. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. Scientific American 284(5), 34–43 (May 2001)
3. Bhagat, S., Cormode, G., Muthukrishnan, S.: Node classification in social networks. In: Aggarwal, C.C. (ed.) Social Network Data Analytics, pp. 115–148. Springer (2011)

4. Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: DBpedia - a crystallization point for the Web of Data. J. Web Sem. 7(3), 154–165 (2009)
5. Chapelle, O., Schölkopf, B., Zien, A. (eds.): Semi-Supervised Learning. MIT Press (2006)
6. Hastie, T., Tibshirani, R., Friedman, J.: The elements of statistical learning: data mining, inference and prediction. Springer, 2 edn. (2008)
7. Heath, T., Bizer, C.: Linked Data: Evolving the Web into a Global Data Space. Synthesis Lectures on the Semantic Web, Morgan & Claypool Publishers (2011)
8. Hellmann, S., Lehmann, J., Auer, S.: Learning of OWL Class Descriptions on Very Large Knowledge Bases. Int. J. Semantic Web Inf. Syst. 5(2), 25–48 (2009)
9. Kelner, J.A., Orecchia, L., Sidford, A., Zhu, Z.A.: A simple, combinatorial algorithm for solving sdd systems in nearly-linear time. In: Boneh, D., et al. (eds.) Proceedings of STOC'13. pp. 911–920. ACM (2013)
10. Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles and Techniques. MIT Press (2009)
11. Lin, H.T., Koul, N., Honavar, V.: Learning Relational Bayesian Classifiers from RDF Data. In: Aroyo, L., et al. (eds.) International Semantic Web Conference (1). LNCS, vol. 7031, pp. 389–404. Springer (2011)
12. Lösch, U., Bloehdorn, S., Rettinger, A.: Graph kernels for RDF data. In: Simperl, E., et al. (eds.) Proceedings of ESWC'12. LNCS, vol. 7295, pp. 134–148. Springer (2012)
13. Minervini, P., d'Amato, C., Fanizzi, N., Esposito, F.: Transductive inference for class-membership propagation in web ontologies. In: Cimiano, P., et al. (eds.) Proceedings of ESWC'13. LNCS, vol. 7882, pp. 457–471. Springer (2013)
14. Nickel, M., Tresp, V., Kriegel, H.P.: A Three-Way Model for Collective Learning on Multi-Relational Data. In: Getoor, L., et al. (eds.) Proceedings of ICML'11. pp. 809–816. Omnipress (2011)
15. Rettinger, A., Lösch, U., Tresp, V., d'Amato, C., Fanizzi, N.: Mining the Semantic Web: Statistical learning for next generation knowledge bases. Data Min. Knowl. Discov. 24(3), 613–662 (2012)
16. Rettinger, A., Nickles, M., Tresp, V.: Statistical Relational Learning with Formal Ontologies. In: Buntine, W.L., et al. (eds.) Proceedings of ECML/PKDD'09. LNCS, vol. 5782, pp. 286–301. Springer (2009)
17. Sirin, E., Parsia, B.: SPARQL-DL: SPARQL Query for OWL-DL. In: Golbreich, C., et al. (eds.) OWLED. CEUR Workshop Proceedings, vol. 258. CEUR-WS.org (2007)
18. Spielman, D.A.: Algorithms, Graph Theory, and Linear Equations in Laplacian Matrices. In: Proceedings of ICM'10. pp. 2698–2722 (2010)
19. Tresp, V., Huang, Y., Bundschus, M., Rettinger, A.: Materializing and querying learned knowledge. In: Proceedings of IRMLeS'09 (2009)
20. Vapnik, V.N.: Statistical learning theory. Wiley, 1 edn. (Sep 1998)
21. de Vries, G.K.D.: A Fast Approximation of the Weisfeiler-Lehman Graph Kernel for RDF Data. In: Blockeel, H., et al. (eds.) ECML/PKDD (1). LNCS, vol. 8188, pp. 606–621. Springer (2013)
22. Zhu, X., Ghahramani, Z., Lafferty, J.D.: Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions. In: Fawcett, T., et al. (eds.) Proceedings of ICML'03. pp. 912–919. AAAI Press (2003)